

SQL SERVER STORED PROCEDURE

S.Mehtap İZMİRLİ AYAN

- Stored Procedure nedir?
- Nasıl hazırlanır?
- Stored Procedure güncelleme.
- Stored Procedure çağırmak.

Stored Procedure Nedir?

- Bir veya daha fazla TSQL (Transact SQL) ifadesinden oluşan bir koleksiyondur.
- Çok kullanılan işlevlerin her defasında yeniden yazmak yerine bu ifadeleri bir yordam altında toplayabilirler.
- Gerektiğinde bu yordam çağrılarak bu işlevler çalıştırılabilir.

Avantajları – 1

- Yordam **bir kez** oluşturulduktan sonra **defalarca** çağrılabilir. Yaptığınız programın kodlarıyla oynamadan sql server da saklı yordam üzerinde değişiklik yapabilirsiniz. Bu işleme **modülerlik** adı verilir.

Avantajları – 2

- Eğer yapılacak işlemler **fazla T-Sql kodu içeriyorsa** ve/veya bu işlemler **tekrar tekrar yapılacaksa** saklı yordam yığın işlemlerinden (go ile ayrılan işlem blokları) çok daha hızlı çalışır. Saklı yordam ilk kez çalıştırıldığında yazım denetimi yapılır ve hız için en iyi seviyeye getirilir. Saklı yordam derlendikten sonra sonraki kullanımlar için hafızada saklanır. Yani her kullanım için yazım denetimi ve hız için en iyi seviyeye getirme ayarı yapılmaz.

Avantajları – 3

- **Ağ trafiğini azaltır.** Bir döngü içerisinde yüzlerce satırın herbirinde baştan sona işlem/işlemler yapılacaksa ağın trafik yoğunluğu artar. Bu durumda döngü işlemleri saklı yordam içerisinde yapılarak ağ trafiği azaltılır.

Avantajları – 4

- Güvenlik
- Saklı Yordamı çalıştırabilmek için kullanıcının izne ihtiyacı vardır. İzne sahip değilse doğrudan saklı yordamı çalıştıramazlar.

Örnekler:

Parametresiz bir örnek;

```
CREATE PROCEDURE stp_IlkOnOgrenci  
AS  
    SET NOCOUNT ON  
    SELECT TOP 10 ogrNo, tcNo, ad, soyad FROM tbl_Ogrenci  
    ORDER BY ogrNo asc
```

Stored Procedure Oluşturulması

Create procedure ILANLARI_GETIR

As

Begin

select

MA.marka_adi as Marka,

MO.model_adi as Model,

IL.eklenme_tarihi as Eklenme

from ilanlar as IL

inner join markalar as MA on MA.marka_id=IL.marka

inner join modeller as MO on MO.model_id=IL.model

Stored Procedure Oluşturulması

Alter procedure ILANLARI_GETIR

As

Begin

select

MA.marka_adi as Marka,

MO.model_adi as Model,

IL.eklenme_tarihi as Eklenme

from ilanlar as IL

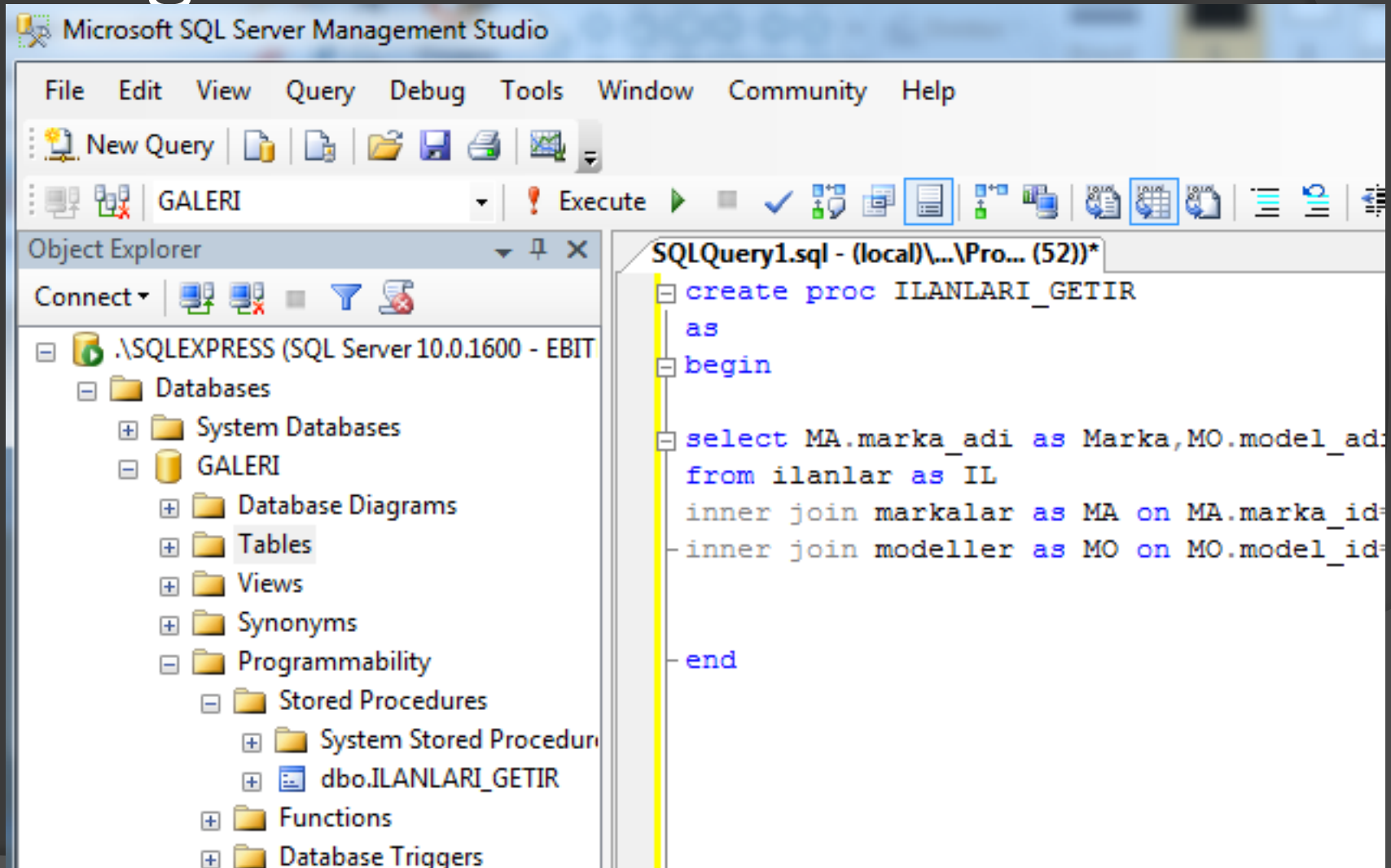
inner join markalar as MA on MA.marka_id=IL.marka

inner join modeller as MO on MO.model_id=IL.model

NO COUNT ON & OFF

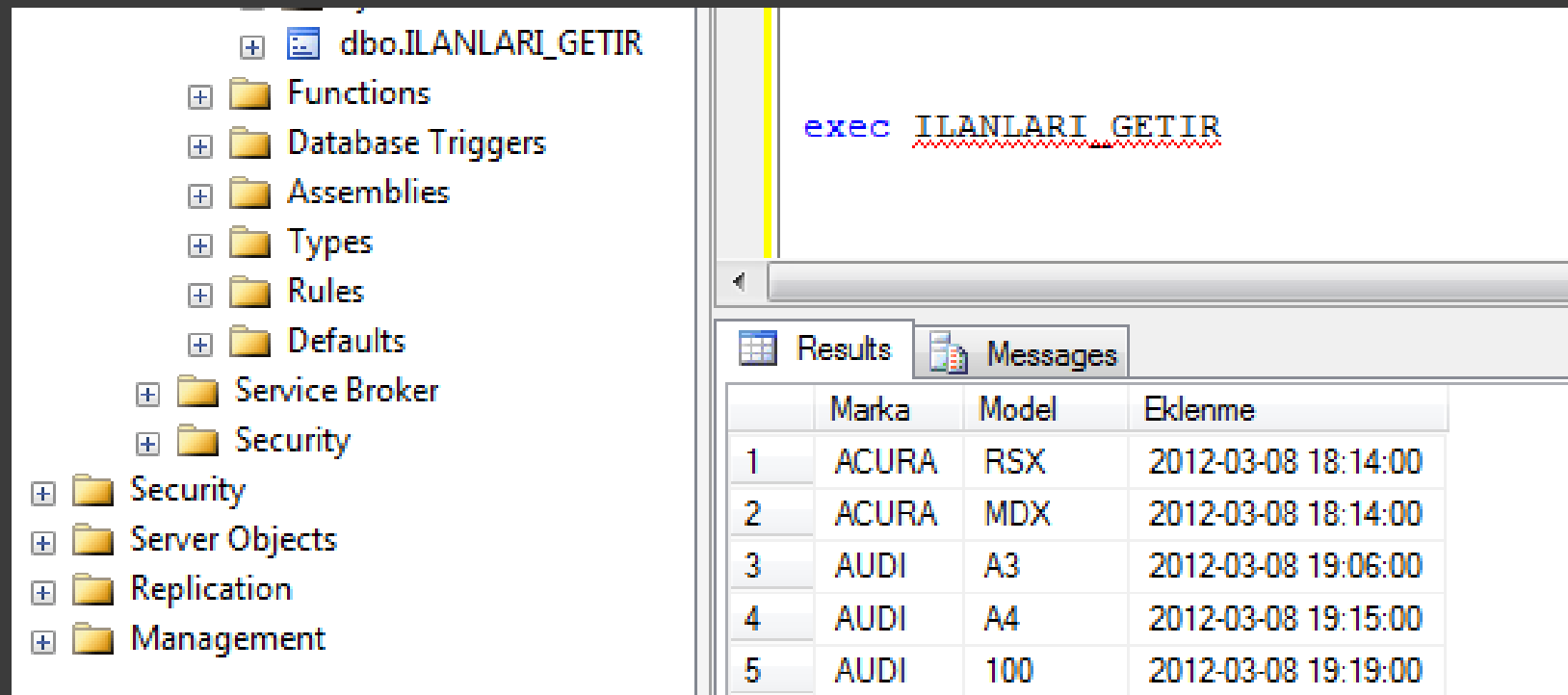
- NOCOUNT oturum parametresi, SQL Server da bir performans artış unsuru olarak kullanılabilir. Çünkü SQL Server her işlemten sonra etkilenen kayıt sayısını mesaj olarak döndürmek üzere fazladan bir döngü çalıştırılacak şekilde ayarlanmıştır. Bu oturum parametresi açıldığında kaç kayıtın etkilendiği sonuç olarak döndürülmez.

Sql Server Management Programındaki Görüntü



Stored Procedur'ün Çağırılması

Exec ILANLARI_GETIR



The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Server Objects' tree is expanded, showing the 'dbo' folder with the stored procedure 'ILANLARI_GETIR' highlighted. The right pane shows the execution of the procedure, with the command 'exec ILANLARI_GETIR' entered. Below the command, the 'Results' tab is active, displaying a table with five rows of data. The columns are 'Marka', 'Model', and 'Eklenme'. The data rows are numbered 1 through 5.

	Marka	Model	Eklenme
1	ACURA	RSX	2012-03-08 18:14:00
2	ACURA	MDX	2012-03-08 18:14:00
3	AUDI	A3	2012-03-08 19:06:00
4	AUDI	A4	2012-03-08 19:15:00
5	AUDI	100	2012-03-08 19:19:00

Parametre Kullanımı

- Saklı yordam içerisindeki kodlar başka programlardan gönderilen parametrelere göre çalıştırılabilir.
- Saklı Yordamlar fonksiyonlar gibi değer döndürebilir, bu döndürülecek değer işlemi parametreye OUTPUT yazısı eklenerek belirtilir.
- Saklı yordamlar 2100 tane parametreye sahip olabilir
- Parametre; parametre adı, veri türü, yönü (output veya input), ve varsayılan değer özelliklerine sahiptir.

Parametre Türleri

- ⦿ input
- ⦿ output
- ⦿ Input-output
- ⦿ return value

Parametrelili bir örnek

```
CREATE PROCEDURE stp_OgrenciBilgi
    @ogrNo NVARCHAR(10)
AS
    SELECT ogrNo,ad,soyad FROM tbl_Ogrenci WHERE ogrNo = @ogrNo
```

`exec stp_OgrenciBilgi '11760101'` --Bir yordamı çalıştırmak için `exec` veya `execute` komutu kullanılır yordam çalıştırıldığında '11760101' nolu öğrenci bilgileri ekrana getirilir.

Parametrelili Procedure Hazırlamak

Create proc ILANLARI_GETIR

(@ilan_id int)

as

Begin

select

MA.marka_adi as Marka,

MO.model_adi as Model, IL.eklenme_tarihi as Eklenme

from ilanlar as IL

inner join markalar as MA on MA.marka_id=IL.marka

inner join modeller as MO on MO.model_id=IL.model

where IL.ilan_id=@ilan_id

end

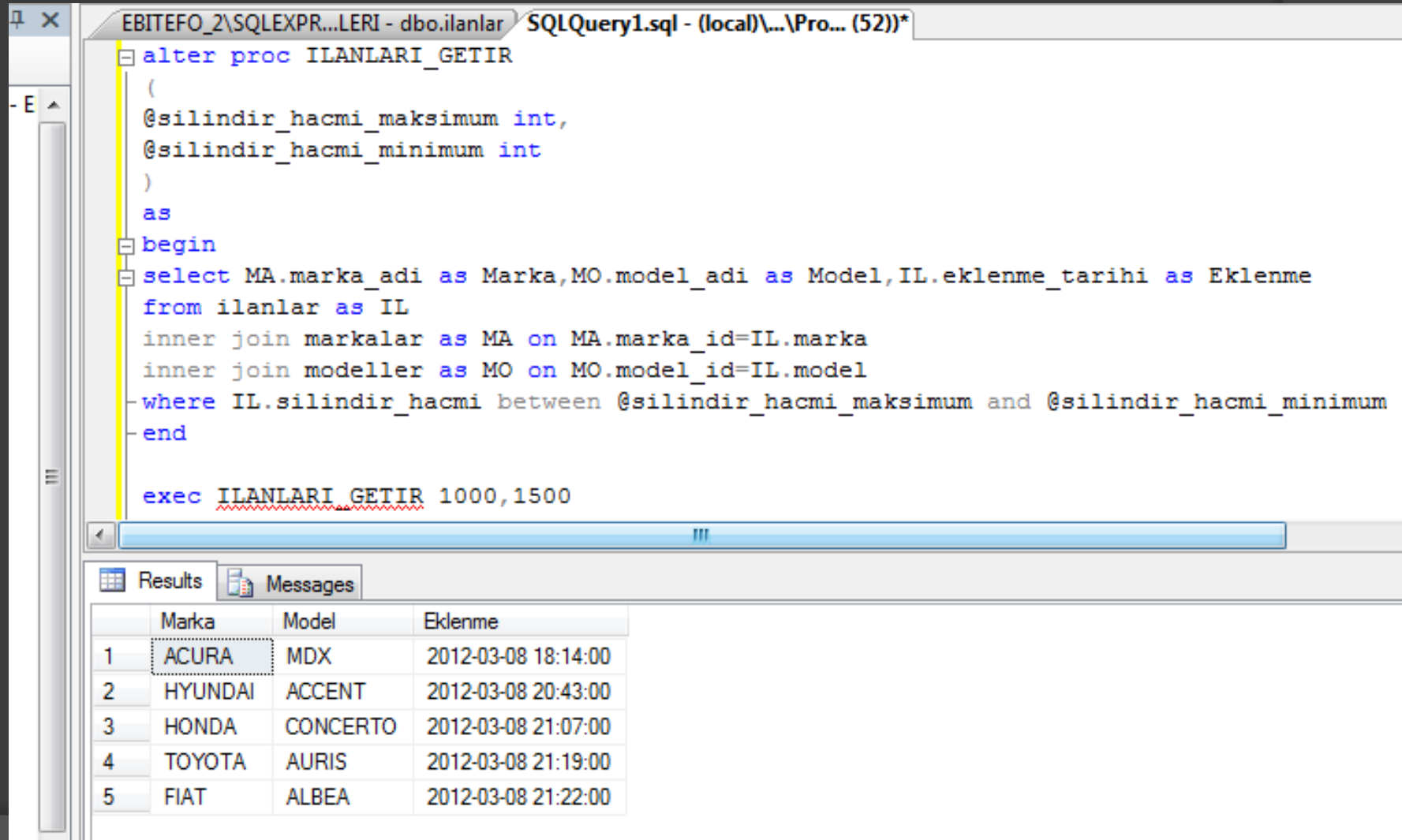
Parametrelili Procedure Çağırılması

The screenshot displays the SQL Server Enterprise Manager interface. On the left, the Object Explorer shows the database structure for 'GALERI', with 'Stored Procedures' expanded to show 'dbo.ILANLARI_GETIR'. The main window shows a SQL query in 'SQLQuery1.sql' that calls the 'ILANLARI_GETIR' procedure with the parameter '2'. The 'Results' tab at the bottom shows the output of the query, which is a table with three columns: 'Marka', 'Model', and 'Eklenme'.

```
SQLQuery1.sql - (local)\...\\Pro... (52))*  
alter proc ILANLARI_GETIR  
(  
    @ilan_id int  
)  
as  
begin  
    select MA.marka_adi as Marka, MO.model  
    from ilanlar as IL  
    inner join markalar as MA on MA.marka  
    inner join modeller as MO on MO.model  
    where IL.ilan_id=@ilan_id  
end  
  
exec ILANLARI_GETIR 2
```

	Marka	Model	Eklenme
1	ACURA	MDX	2012-03-08 18:14:00

Birden fazla parametre kullanımı



The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows the definition of a stored procedure named `ILANLARI_GETIR`. The procedure takes two integer parameters, `@silindir_hacmi_maksimum` and `@silindir_hacmi_minimum`, and returns a result set with columns `Marka`, `Model`, and `Eklenme`. The procedure body includes a `select` statement with `inner join` clauses for `markalar` and `modeller`, and a `where` clause filtering by the cylinder capacity range. The bottom pane shows the results of executing the procedure with parameters 1000 and 1500, displaying a table with 5 rows of car data.

```
alter proc ILANLARI_GETIR
(
    @silindir_hacmi_maksimum int,
    @silindir_hacmi_minimum int
)
as
begin
    select MA.marka_adi as Marka, MO.model_adi as Model, IL.eklenme_tarihi as Eklenme
    from ilanlar as IL
    inner join markalar as MA on MA.marka_id=IL.marka
    inner join modeller as MO on MO.model_id=IL.model
    where IL.silindir_hacmi between @silindir_hacmi_maksimum and @silindir_hacmi_minimum
end

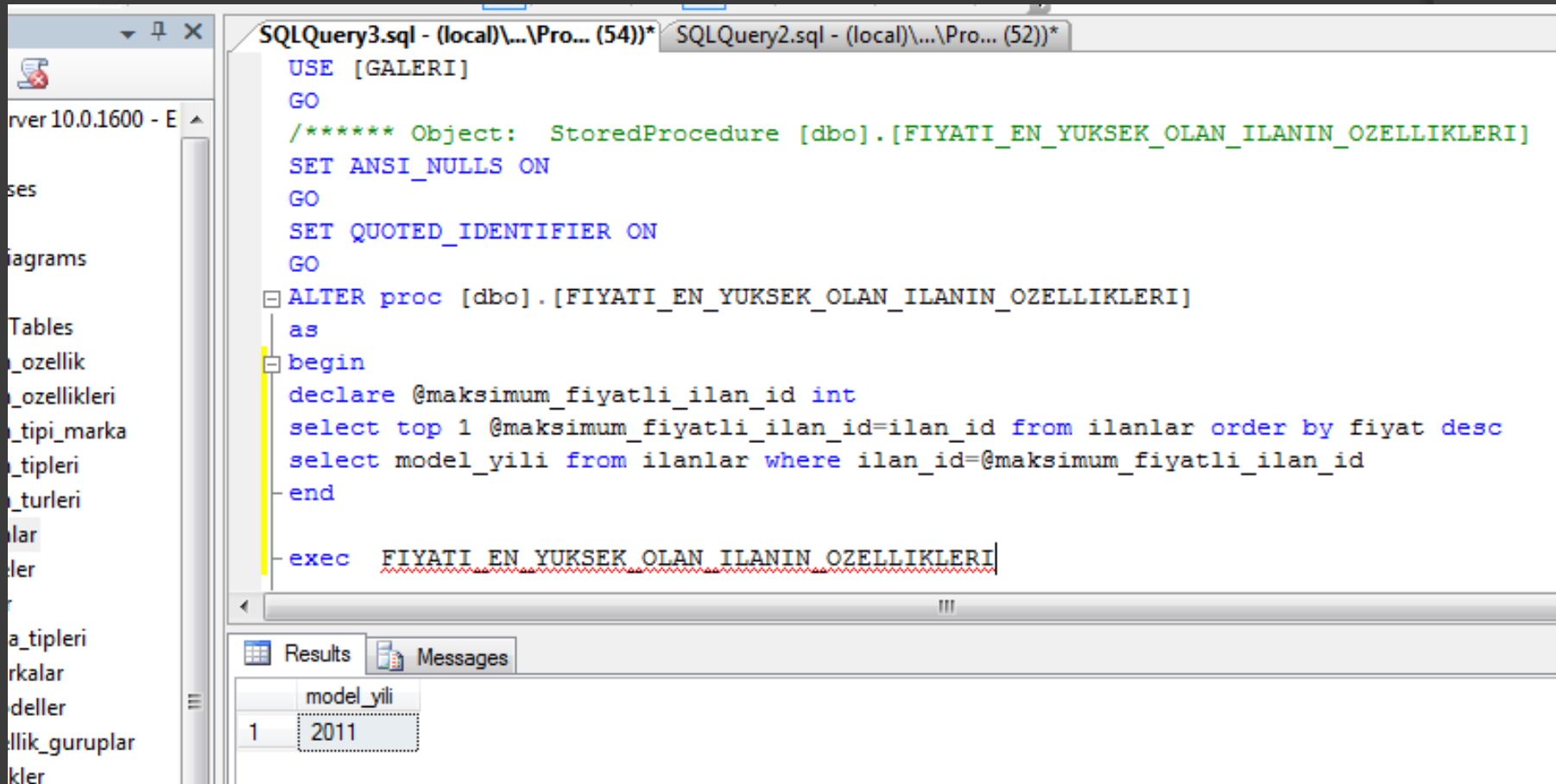
exec ILANLARI_GETIR 1000,1500
```

	Marka	Model	Eklenme
1	ACURA	MDX	2012-03-08 18:14:00
2	HYUNDAI	ACCENT	2012-03-08 20:43:00
3	HONDA	CONCERTO	2012-03-08 21:07:00
4	TOYOTA	AURIS	2012-03-08 21:19:00
5	FIAT	ALBEA	2012-03-08 21:22:00

Prosedürün Silinmesi

```
drop proc ILANLARI_GETIR
```

Prosedür içerisinde değişken tanımlanması



The screenshot displays the SQL Server Enterprise Manager interface. The left pane shows the database structure for 'GALERI' (10.0.1600 - E), including tables like 'ozellik' and 'ozellikleri'. The main pane shows the SQL script for 'SQLQuery3.sql' (local)\...\Pro... (54))*.

```
USE [GALERI]
GO
/***** Object:  StoredProcedure [dbo].[FIYATI_EN_YUKSEK_OLAN_ILANIN_OZELLIKLERI]
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER proc [dbo].[FIYATI_EN_YUKSEK_OLAN_ILANIN_OZELLIKLERI]
as
begin
declare @maksimum_fiyatli_ilan_id int
select top 1 @maksimum_fiyatli_ilan_id=ilan_id from ilanlar order by fiyat desc
select model_yili from ilanlar where ilan_id=@maksimum_fiyatli_ilan_id
end

exec FIYATI_EN_YUKSEK_OLAN_ILANIN_OZELLIKLERI
```

The bottom pane shows the 'Results' tab with a single row of data:

	model_yili
1	2011